

<i>RESPONSE C</i>	
Appl. No.: 09/992,121 Office action Dated: 11/16/2006 Response Dated: 05/14/07	

Listing of Claims

This listing of claims will replace all prior versions, and listings of claims in the application.

1. (Currently Amended) A computer-implemented method for dynamic emulation of legacy instructions comprising:

accessing said legacy instructions in legacy blocks, each legacy block including a plurality of legacy instructions,

for each particular legacy instruction in a particular legacy block,

translating the particular legacy instruction into one or more particular translated instructions for emulating the particular legacy instruction, organizing the particular translated instructions into one or more particular translated blocks,

linking the particular translated blocks into a particular linked group corresponding to said particular legacy block; said linking using a link in each particular translated block to point to a location of the next particular translated block of the particular linked group,

executing the particular translated instructions in the particular translated blocks by ~~sequentially~~ executing the linked group translated blocks.

2. (Cancelled)
3. (ORIGINAL) The method of Claim 1 wherein said particular translated instructions are stored in a cache and wherein said particular translated instructions are purged from said cache only when all said particular translated instructions of particular translated blocks are also purged from said cache.

<i>RESPONSE C</i>	
Appl. No.: 09/992,121 Office action Dated: 11/16/2006 Response Dated: 05/14/07	

4. (ORIGINAL) The method of Claim 1 wherein said legacy instructions are for a legacy system having a S/390 architecture.
5. (ORIGINAL) The method of Claim 1 wherein said legacy instructions are object code instructions compiled/assembled for a legacy architecture.
6. (ORIGINAL) The method of Claim 1 wherein said translated instructions are for execution in a RISC architecture.

<i>RESPONSE C</i>	
Appl. No.: 09/992,121 Office action Dated: 11/16/2006 Response Dated: 05/14/07	

7. (Currently Amended) A computer-implemented method for dynamic emulation of legacy instructions, where the legacy instructions are compiled/assembled into object code form for a native architecture, where the legacy instructions are executed as guests in the host architecture, where the legacy instructions are translated to translated instructions in the host architecture and the translated instructions are executed in the host architecture concurrently with the translation of the legacy instructions in the host architecture, comprising:

accessing said legacy instructions in legacy blocks of a host system operating with said host architecture, each legacy block including a plurality of legacy instructions,
for each particular legacy instruction in a particular legacy block,

translating the particular legacy instruction into one or more particular translated instructions of the host system for emulating the particular legacy instruction as a guest in said host architecture,

organizing the particular translated instructions into one or more particular translated blocks,

linking the particular translated blocks into a particular linked group corresponding to said particular legacy block; said linking using a link in each particular translated block to point to a location of the next particular translated block of the particular linked group,

executing the particular translated instructions in the particular translated blocks by ~~sequentially~~ executing the linked group translated blocks.

<i>RESPONSE C</i>	
Appl. No.: 09/992,121 Office action Dated: 11/16/2006 Response Dated: 05/14/07	

8. (New C) A computer-implemented method for dynamic emulation of legacy instructions, where the legacy instructions are compiled/assembled into object code form for a native architecture, where the legacy instructions are executed as guests in the host architecture, where the legacy instructions are translated to translated instructions in the host architecture and the translated instructions are executed in the host architecture concurrently with the translation of the legacy instructions in the host architecture, comprising:

accessing said legacy instructions in legacy blocks of a host system operating with said host architecture, each legacy block including a plurality of contiguous legacy instructions,

for each particular legacy instruction in a particular legacy block,

translating the particular legacy instruction into one or more particular translated instructions of the host system for emulating the particular legacy instruction as a guest in said host architecture,

organizing the particular translated instructions into one or more particular translated blocks, each translated block including a plurality of contiguous translated instructions stored in a cache,

linking the particular translated blocks in said cache into a logical group corresponding to said particular legacy block; said logical group including a first translated block, one or more next translated blocks and a last block, said linking using a linked list in said cache including a first link in the first translated block that points to a location in the cache of a next translated block, one or more next links in the next translated blocks where each next link points to a location in the cache of a subsequent one of the next translated blocks, and a last link that points to the last block of the logical group,

<i>RESPONSE C</i>	
Appl. No.: 09/992,121 Office action Dated: 11/16/2006 Response Dated: 05/14/07	

executing the particular translated instructions in the particular translated blocks of the logical group.

9. (New C) The method of Claim 8 where said plurality of contiguous legacy instructions in the legacy block include one or more legacy branch instructions, where the translating step translates the legacy branch instructions to translated branch instructions, where the executing step executes said translated branch instructions and when a taken branch results in a branch target instruction within the logical group, the executing step directly executes the branch target instruction without requiring an external reference to look-up the location of the branch target instruction.
10. (New C) The method of Claim 8 where each logical group of translated blocks corresponds to at least one legacy block.
11. (New C) The method of Claim 8 where each legacy block has a number of translated blocks where the number of translated blocks differs for different legacy blocks.
12. (New C) The method of Claim 8 wherein said legacy instructions are for a legacy system having a S/390 architecture.
13. (New C) The method of Claim 8 wherein said legacy instructions are object code instructions compiled/assembled for a legacy architecture.
14. (New C) The method of Claim 8 wherein said translated instructions are for execution in a RISC architecture.